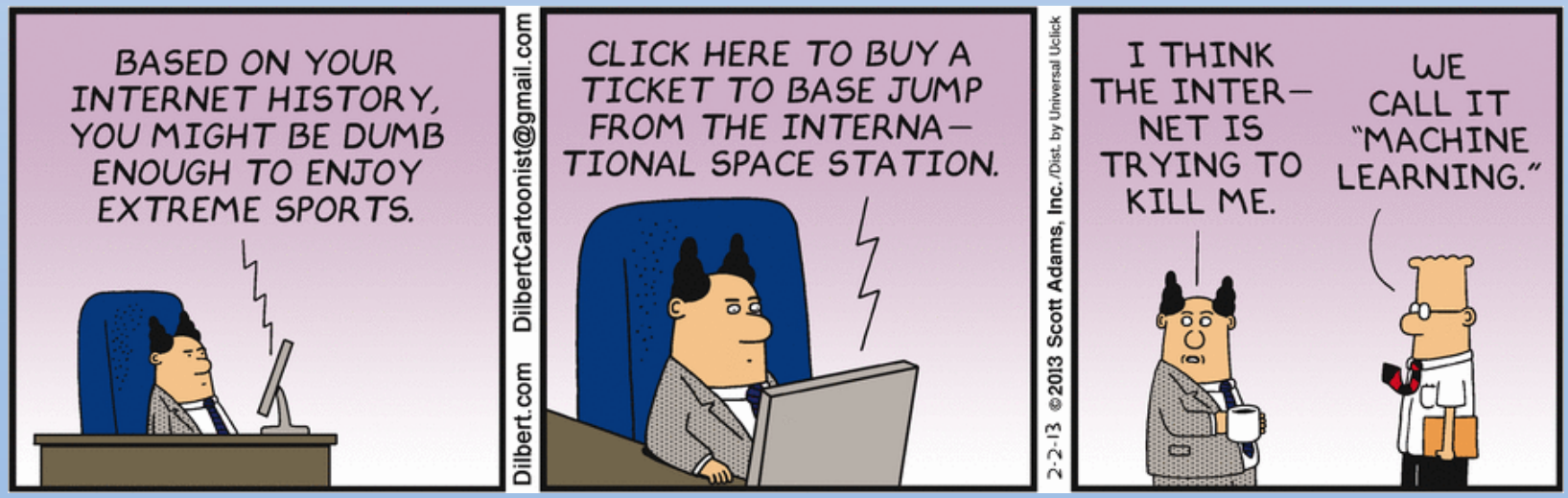
Dr. Ali Arsanjani
@AliArsanjani
AI/ML Specialist
AWS
---
Ex-VP AI and Machine Learning,
DeepContext,
Ex-IBM Distinguished Engineer, CTO Analytics and Machine Learning
---
Adjunct At San Jose State University
Senior Lecturer UCSD

# Applied Machine Learning

## Session 1


(c) Ali Arsanjani, 2019




Source: Dilbert.com

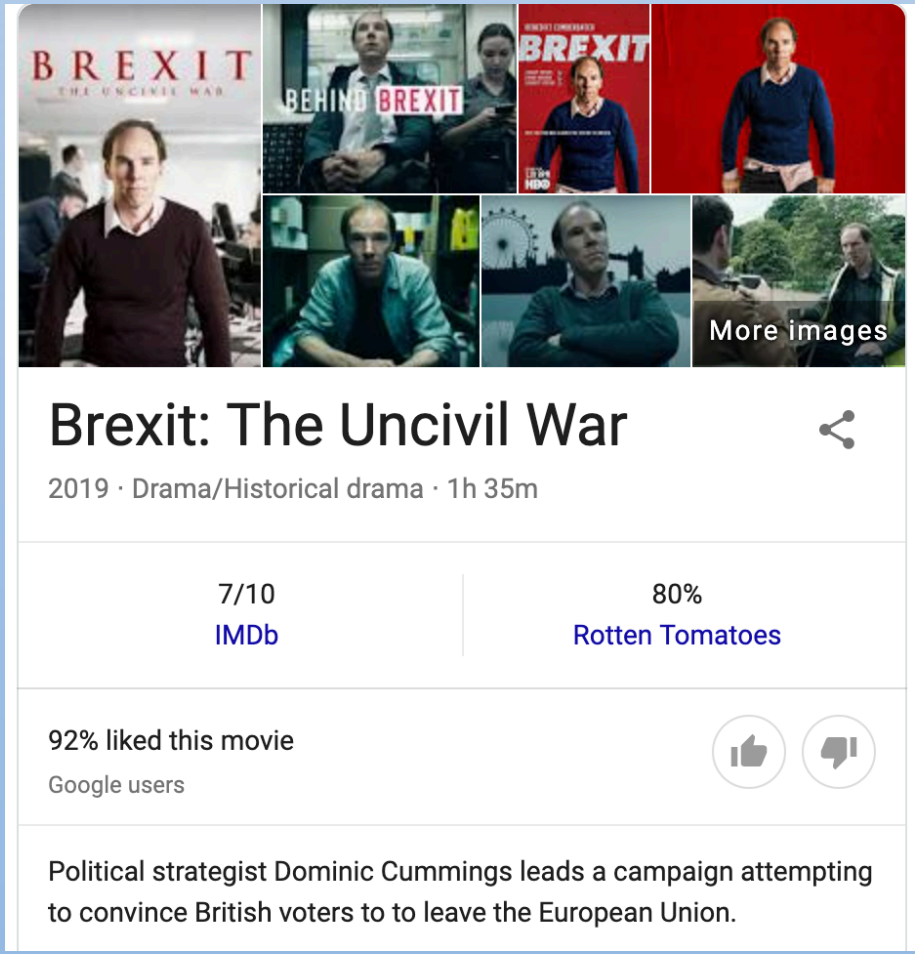
# Class Objectives

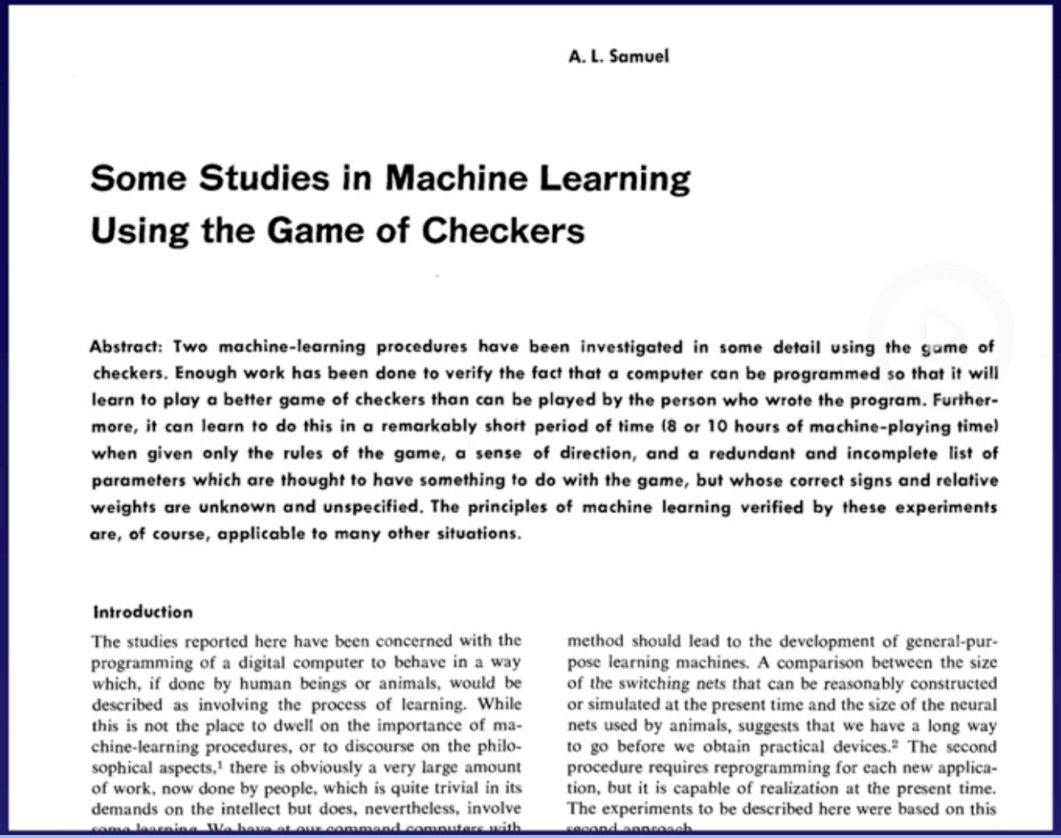
- Understand and apply the foundational aspects of machine learning
  - In a holistic manner
  - In conjunction with software engineering best practices
  - Learn ML best practices and worst practices
- Have fun learning the coolest and most promising technology yet
- Gain insight into the social, ethical implications of AI and ML

- Output : data scientist, machine learning engineer
  - + research oriented + focus on applications
- Input(pre-reqs) : dogged persistence + linear algebra + matrix math + teeny bit of calculus + python programming + collaborative attitude

# We have burden of responsibility as ML Engineers and Data Scientists … our work can have global impact

- How data science affected the UK's Brexit





A. L. Samuel

## Some Studies in Machine Learning Using the Game of Checkers

**Abstract:** Two machine-learning procedures have been investigated in some detail using the game of checkers. Enough work has been done to verify the fact that a computer can be programmed so that it will learn to play a better game of checkers than can be played by the person who wrote the program. Furthermore, it can learn to do this in a remarkably short period of time (8 or 10 hours of machine-playing time) when given only the rules of the game, a sense of direction, and a redundant and incomplete list of parameters which are thought to have something to do with the game, but whose correct signs and relative weights are unknown and unspecified. The principles of machine learning verified by these experiments are, of course, applicable to many other situations.

### Introduction

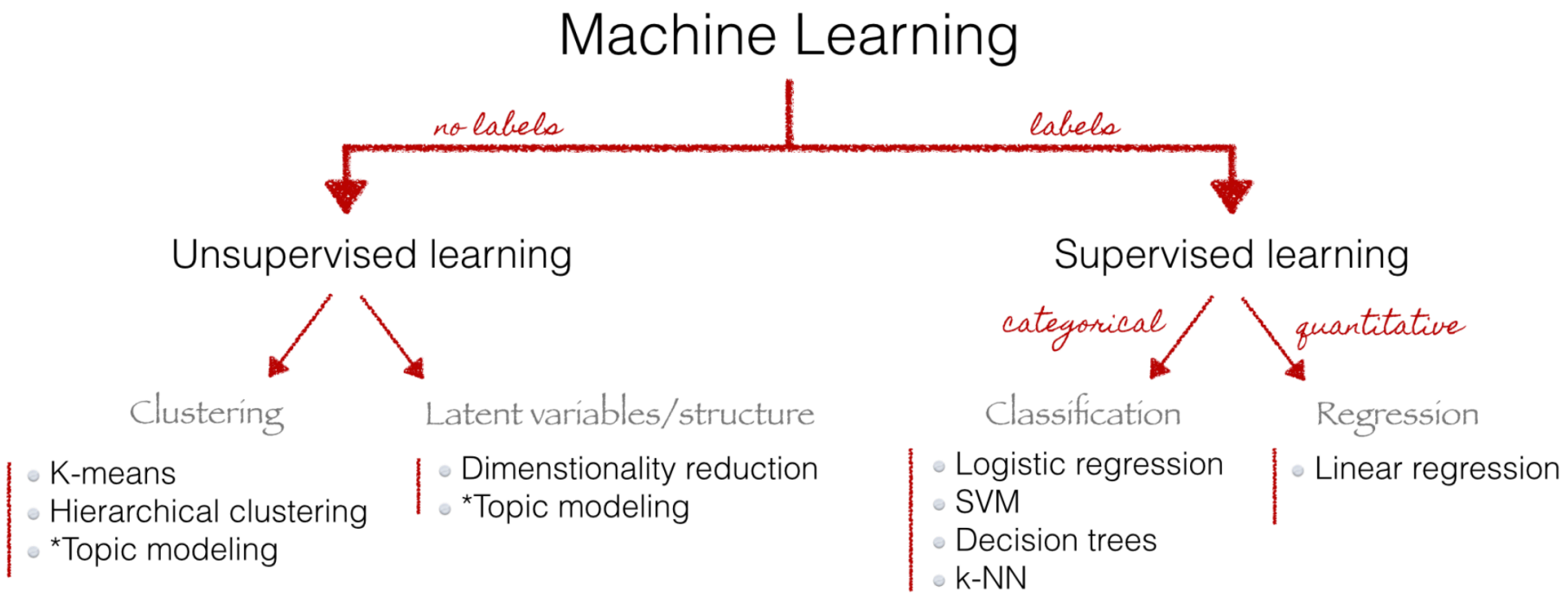
The studies reported here have been concerned with the programming of a digital computer to behave in a way which, if done by human beings or animals, would be described as involving the process of learning. While this is not the place to dwell on the importance of machine-learning procedures, or to discourse on the philosophical aspects,[1] there is obviously a very large amount of work, now done by people, which is quite trivial in its demands on the intellect but does, nevertheless, involve some learning. We have at our command computers with

method should lead to the development of general-purpose learning machines. A comparison between the size of the switching nets that can be reasonably constructed or simulated at the present time and the size of the neural nets used by animals, suggests that we have a long way to go before we obtain practical devices.[2] The second procedure requires reprogramming for each new application, but it is capable of realization at the present time. The experiments to be described here were based on this second approach.

# Arthur Samuel

IBM Journal 1959

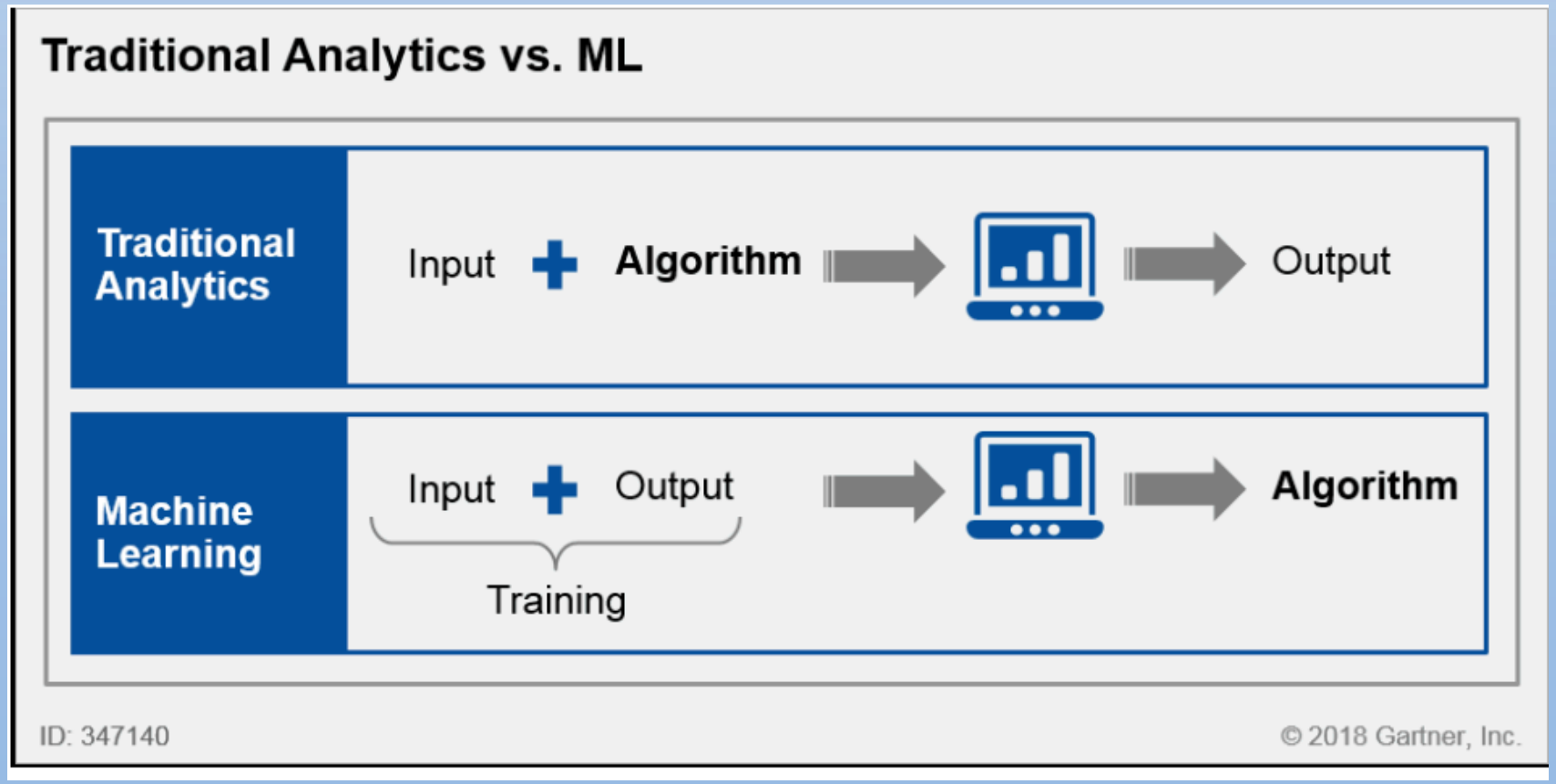
"To verify the fact that a computer be programmed so that it will learn to play a better game of checkers than can be played by the person who wrote the program?"


Machine Learning
no labels
labels
Unsupervised learning
Supervised learning
categorical
quantitative
Clustering
Latent variables/structure
Classification
Regression
K-means
Hierarchical clustering
*Topic modeling
Dimenstionality reduction
*Topic modeling
Logistic regression
SVM
Decision trees
k-NN
Linear regression

# Cottage Industry → Industrial Scale

Machine learning and intelligence

We need a process…


Traditional Analytics vs. ML
Traditional Analytics
Input
Algorithm
Output
Machine Learning
Input
Output
Training
Algorithm
ID: 347140
© 2018 Gartner, Inc.

## TAXONOMY OF MACHINE LEARNING METHODOLOGIES

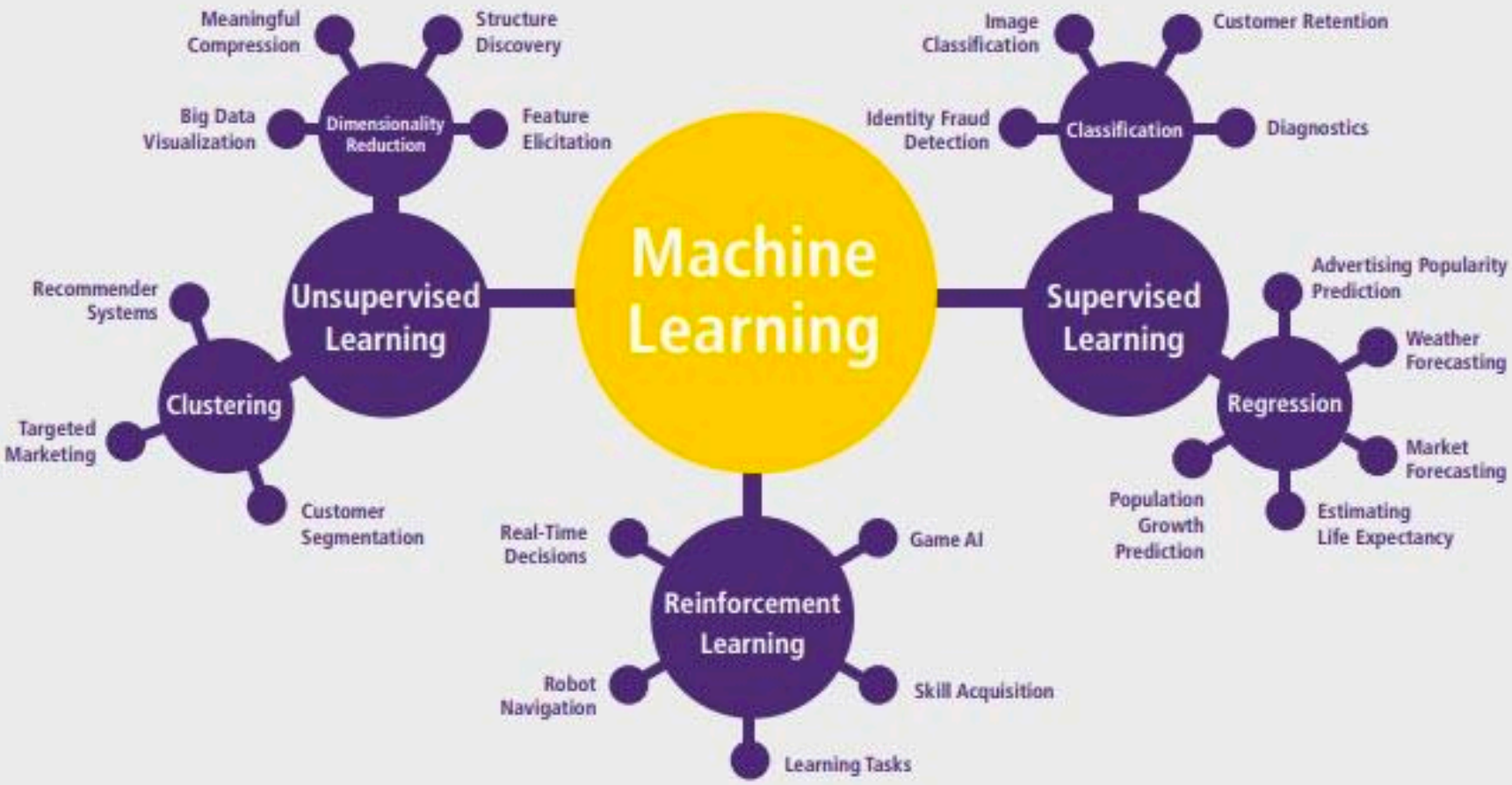



**Figure 10:** An overview of machine learning techniques; **Source:** Jha, V.

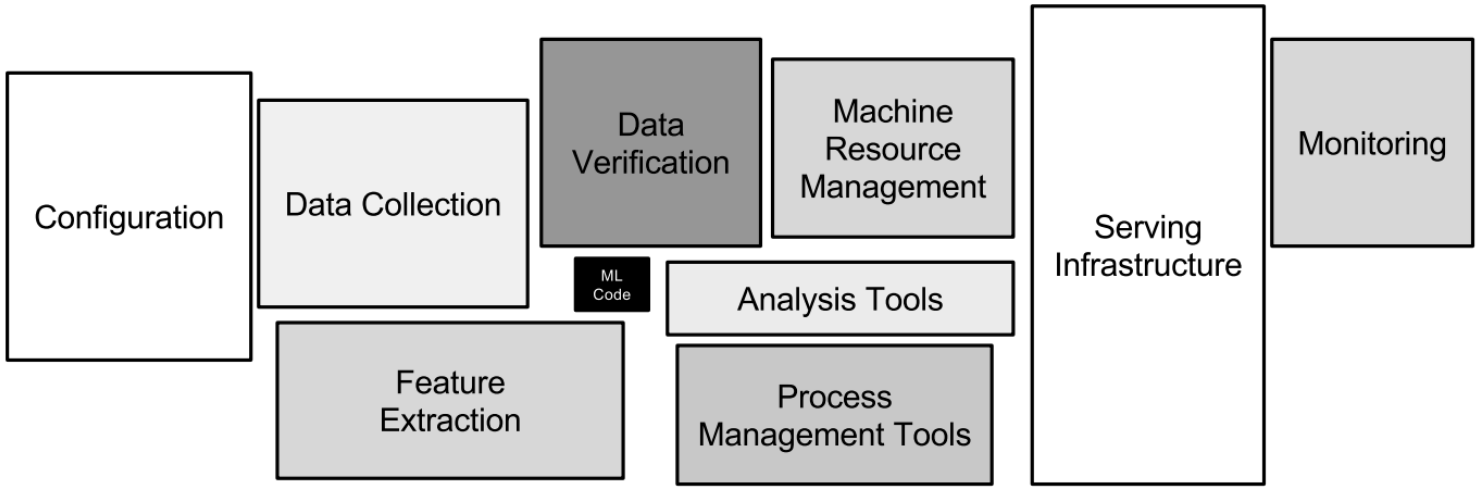



Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

Hidden Technical Debt in Machine Learning Systems

https://papers.nips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf

# Use Big Data driven PREDICTIVE ANALYTICS to Optimize Sales

- Context helps refine and focus advanced predictive analytics solutions to
- prioritize and optimize the greatest potential for high-volume sales, reduce churn and increase relevant upsell

- Organizations can optimize
- Marketing spend and account management to
- increase estimated revenue

# How GM Uses BIG DATA to Generate Sales

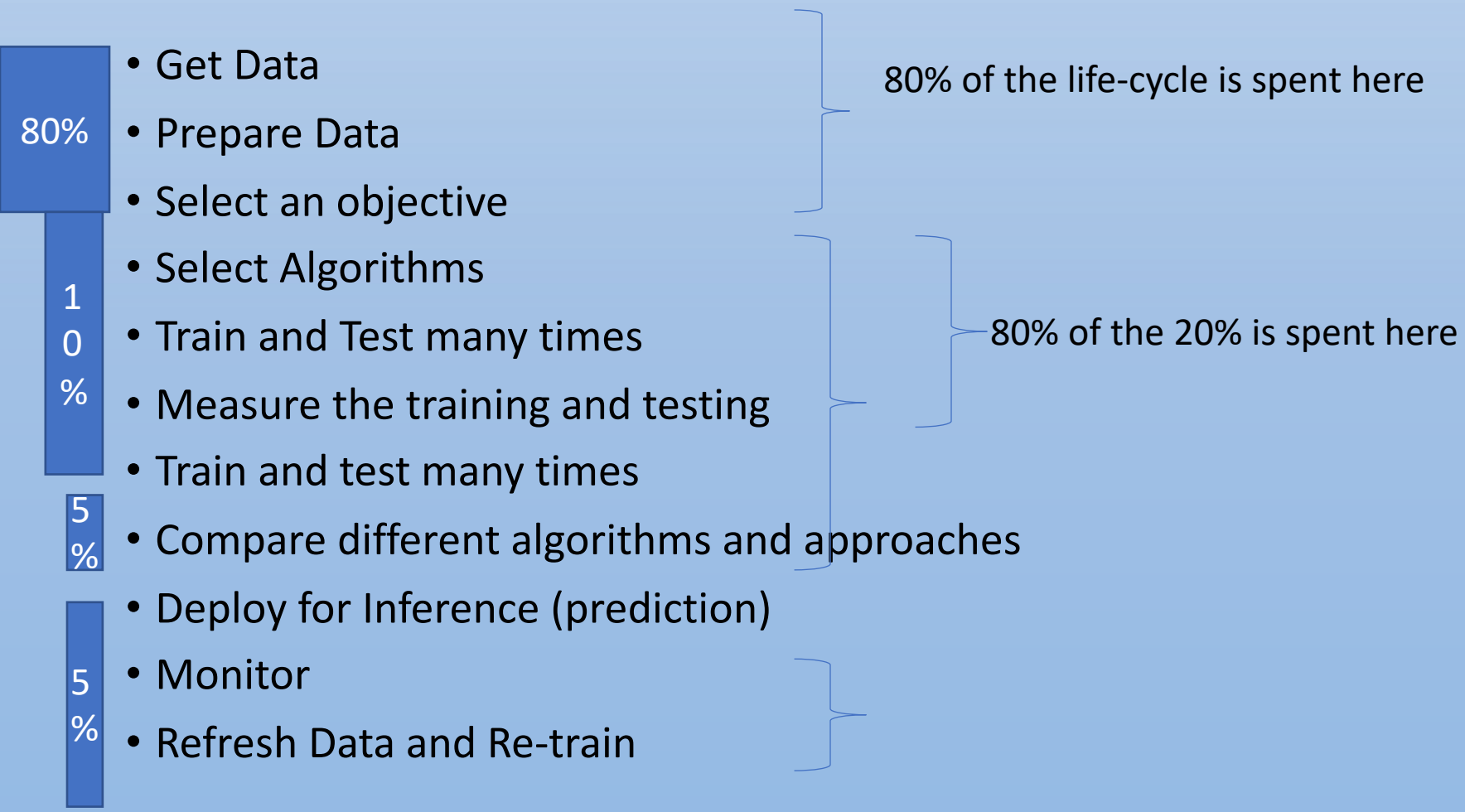
- General Motors combined big data, analytics, and GIS to model dealership performance.
- It enabled dealers from around the nation to view local demographics, location characteristics, and regional differences to providing a dealership the ability to compare their performance to actual results.
- **Driving for a deal**
- Instead of approaching the marketing arena, which GM budgets around $2 billion each year, they *decided to conduct an analyses to determine the types of households that will buy the various automobiles within its portfolio*.
- By feeding detailed demographic and spatial data to marketing, they were able ***direct its ad spend*** to the right departments.
- Leverage DeepContext , “ask me how I remember tomorrow?” ™

# Amalgamation – What Direction is the Data Pointing?

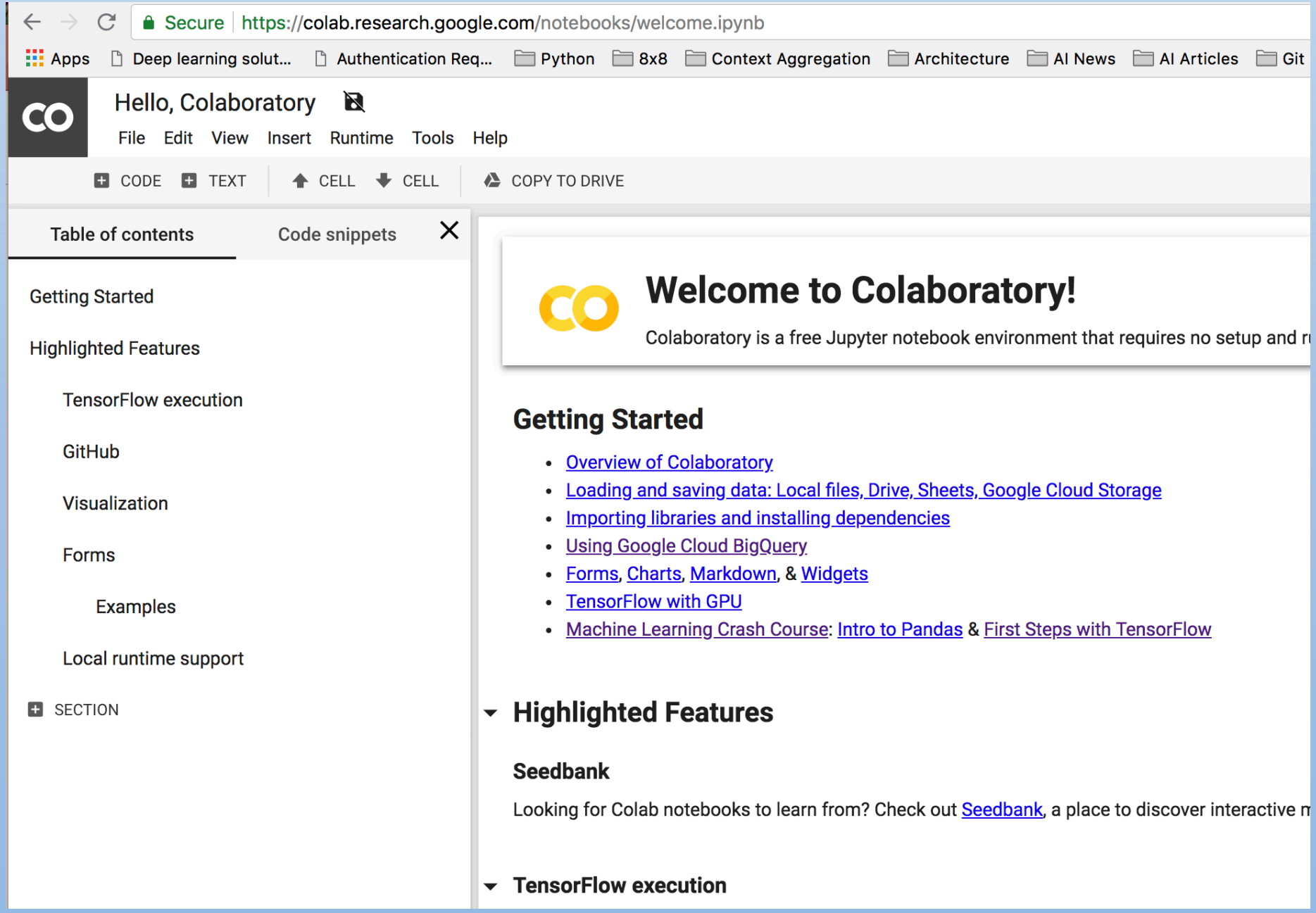
# What do we do in Machine Learning?

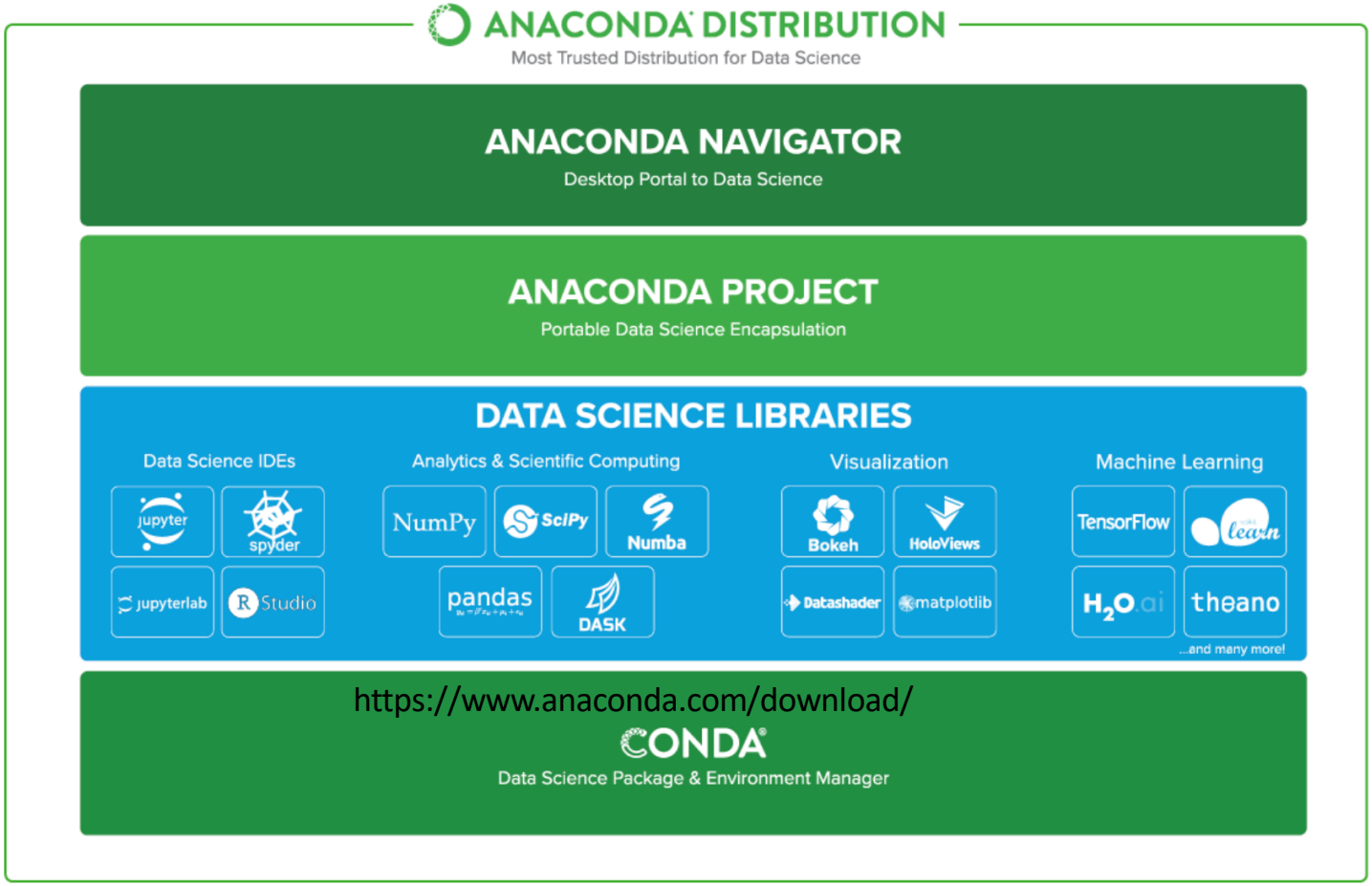
- Clustering
  - Related to segmentation,
  - What are the types of customer we have in this dataset?
  - What are the customer segments I should use in my marketing campaign?
- Classification
  - Is this a picture of a dog or a cat?
  - What are the topics in this email? This article? This book?
  - What is the sentiment of this tweet?
- Regression
  - Given the temperatures of cities on Earth for the past 200 years, can you predict the temperature of San Diego next month?
  - Given past housing pries, can you predict the price of a given house in Chicago?
- Recommendations
  - Collaborative Filtering
  - …prediction

# How do we do Machine Learning?



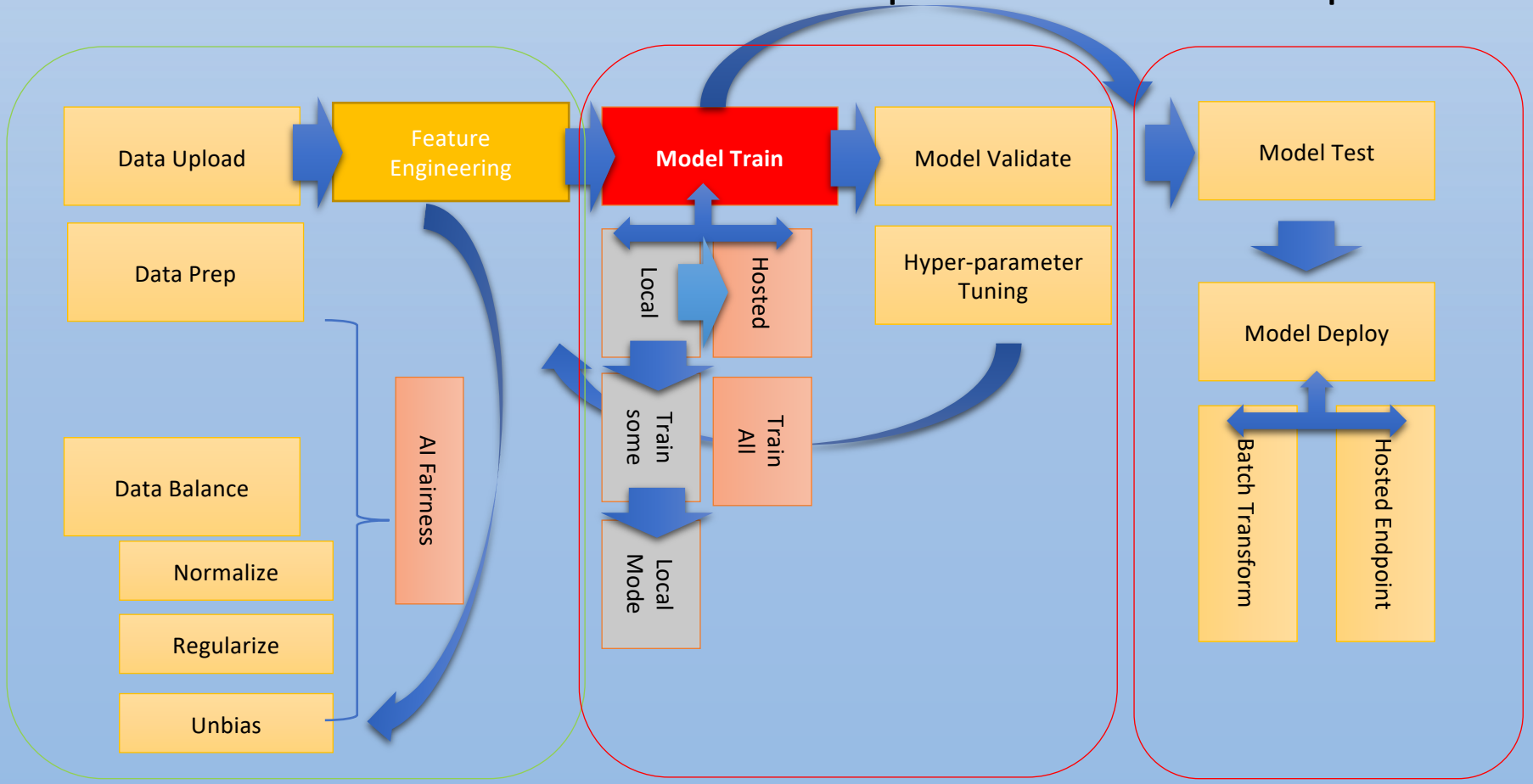
https://colab.research.google.com/notebooks/welcome.ipynb






Why Over 6 Million Users Love Anaconda Distribution

# Recall Some of the Main Steps in the ML Pipeline



# ML: The Universal Function Approximator

$$Y = f(x)$$

Given Y, the outputs of the function f
Given the inputs to the function f
***Can you learn f?***

# Machine learning algorithms

- We supervise the labelling of a dataset
- Given this dataset of labelled "supervised training" data, how can we (think algorithms) make the machine generalize and provide accurate inferences/predictions
- …about a set of previously "unseen", "unknown" or "hidden" (latent) data?
- Machine learning algorithms are designed to implement this idea.
- They use a range of different assumptions and input data types.
- They range from simplistic like K-means clustering to super cool yet complex like Latent Dirichlet Allocation.
  - …And then there are an innumerable number of variations in the middle

# Python 3.x

1

- Write a program to convert Celsius to Fahrenheit
- You have 15 minutes
- Submit your Answer on canvas

$$f = c \times 1.8 + 32$$

2

- Write the same program using numpy arrays with the following data as examples

f(-40.0) = -40.0
f(-10.0) = 14.0
f(0.0) = 32.0
f(8.0) = 46.0
f(15.0) = 59.0
f(22.0) = 72.0
f(38.0) = 100.0

3

## Now, Learn to Convert Celsius to Farenheit

# Learn the function to convert C to F

- import tensorflow as tf
- import numpy as np

# Data Stuff

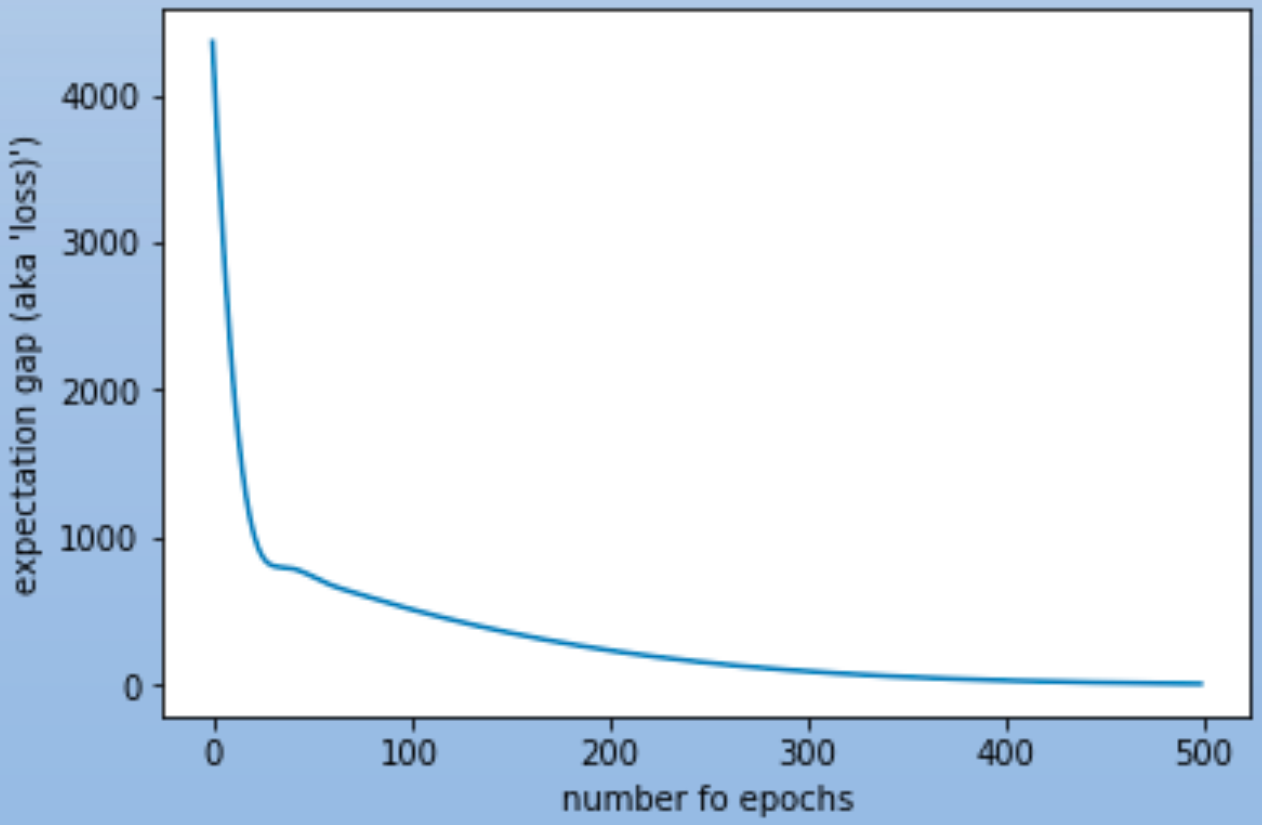
- celcius_data   = np.array([-40, -10,  0,  8, 15, 22,  38],dtype=float)
- farenheit_data = np.array([-40,  14, 32, 46, 59, 72, 100],dtype=float)
- for f,c in enumerate(celcius_data):
-  print("{} C => {} F".format(c,farenheit_data[f]))

# Neural Network Model

- temperature_net = tf.keras.layers.Dense(units = 1,input_shape=[1])
- temperature_learner =tf.keras.models.Sequential([temperature_net])
- trainer = temperature_learner.fit(celcius_data,farenheit_data,epochs=500,verbose=False)

# Measure

- import matplotlib.pyplot as plt
- plt.xlabel("number fo epochs")
- plt.ylabel("expectation gap (aka 'loss)')")
- plt.plot(trainer.history['loss'])



# Test the model : predict

- print(temperature_learner.predict([100.0]))
- >> [[211.30865]]

# Teams

- Team up : teams of [2-4] people
- Choose an area you are passionate about but be prepared to meet in the middle
- Prove the work is enough for 4
- Demonstrate unequivocally how each is contributing every week
- Have each member have a portion to
  - Research
  - Code
  - Design
  - Dig up data sets
  - Curate and scrape data
  - Create a portion of the data narrative
  - Define an objective function to fulfill a business need

The *confusion matrix* shows the ways in which your classification model is confused when it makes predictions.

**insight not only into the errors being made but more importantly the types of errors that are being made.**

**Confusion Matrix and ROC Curve**

| | | Predicted Class | |
|---|---|---|---|
| | | No | Yes |
| Observed Class | No | TN | FP |
| | Yes | FN | TP |

| | |
|---|---|
| TN | True Negative |
| FP | False Positive |
| FN | False Negative |
| TP | True Positive |

**Model Performance**

| | |
|---|---|
| Accuracy | = (TN+TP)/(TN+FP+FN+TP) |
| Precision | = TP/(FP+TP) |
| Sensitivity | = TP/(TP+FN) |
| Specificity | = TN/(TN+FP) |

# Deep Context Life-cycle

Parallel Work streams

1. Data Selection, Ingestion, Preparation, **Curation** (Data **Pipeline**)
   - Entity Event-timeline focus, a la BELA (Business Entity Life-cycle Analysis)
   - Fairness, Accessibility, Transparency; Ethics Governance possible blockchain integration
   - XAI, Interpretability Factors
     - Local or global perturbations or assessments
2. Data **Narrative**.
   - Figure out your data narrative for business alignment and impact: Optimization, Expectation and Loss Functions and use-cases
3. **Algorithms Selection**
   - Algorithms fit for purpose and matching use-cases to fulfill your narrative
4. **Identify Latent Variables**, Models, Manifolds
   - Identify at least one Latent Manifold, which variables should you use? Enrich data in the next step to feed into your latent manifold, to help with explaining results and contributing to your Data Narrative
5. **Perform Data Distillations**
   - Apply relevant distillations
6. **Construct Time lines and context graphs**
   - Construct an event timeline e.g., of the customer behavior, financial txns, patient case, insurance claim, stock symbol relative to market, realestate prices, etc.
7. **Context Amalgamation and Data Enrichment**
   - Apply amalgamation techniques, identify and report back on your amalgamation and how it possibly increased your accuracy, R2, F1 RSME, etc.
8. **enrich Context through amalgamation**
9. **Model Training, Assessment and Tuning**
10. **Model Deployment and Monitoring**


Deep
Context
AI